IMPLEMENTASI FUZZING MENGGUNAKAN AFL++ UNTUK MENEMUKAN KERENTANAN PADA PERANGKAT LUNAK BERBASIS BINARY

SKRIPSI

Diajukan untuk memenuhi salah satu syarat mencapai derajat Sarjana

Program Studi Teknik Komputer

disusun oleh

AHMAD HASSAN RASYIDI

20.83.0577

Kepada

FAKULTAS ILMU KOMPUTER

UNIVERSITAS AMIKOM YOGYAKARTA

YOGYAKARTA

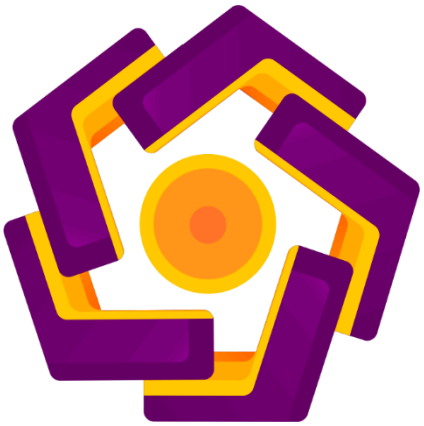
2025

IMPLEMENTASI FUZZING MENGGUNAKAN AFL++ UNTUK MENEMUKAN KERENTANAN PADA PERANGKAT LUNAK BERBASIS BINARY

SKRIPSI

untuk memenuhi salah satu syarat mencapai derajat Sarjana

Program Studi *Teknik Komputer*



disusun oleh

AHMAD HASSAN RASYIDI

20.83.0577

Kepada

FAKULTAS ILMU KOMPUTER

UNIVERSITAS AMIKOM YOGYAKARTA

YOGYAKARTA

2025

HALAMAN PERSETUJUAN

SKRIPSI

IMPLEMENTASI FUZZING MENGGUNAKAN AFL++ UNTUK MENEMUKAN KERENTANAN PADA PERANGKAT LUNAK BERBASIS BINARY

yang disusun dan diajukan oleh

Ahmad Hassan Rasyidi

20.83.0577

telah disetujui oleh Dosen Pembimbing Skripsi
pada tanggal 9 Juni 2025

Dosen Pembimbing,

Muhammad Rudyanto Arief, S.T., M.T.
NIK. 190302098

HALAMAN PENGESAHAN

SKRIPSI

IMPLEMENTASI FUZZING MENGGUNAKAN AFL++ UNTUK MENEMUKAN KERENTANAN PADA PERANGKAT LUNAK BERBASIS BINARY

yang disusun dan diajukan oleh

Ahmad Hassan Rasyidi

20.83.0577

Telah dipertahankan di depan Dewan Penguji
pada tanggal 30 Juni 2025

Susunan Dewan Penguji

Nama Penguji	Tanda Tangan
Joko Dwi Santoso, M. Kom. **NIK. 190302181**	
Ali Mustopa, S.Kom, M.Kom. **NIK. 190302192**	
M. Rudyanto Arief, S.T., M.T. **NIK. 190302098**	

Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Komputer
Tanggal 30 Juni 2025

DEKAN FAKULTAS ILMU KOMPUTER

Prof. Dr. Kusrini, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Yang bertandatangan di bawah ini,

Nama mahasiswa : Ahmad Hassan Rasyidi
NIM : 20.83.0577

Menyatakan bahwa Skripsi dengan judul berikut:

Implementasi Fuzzing Menggunakan AFL++ Untuk Menemukan Kerentanan Pada Perangkat Lunak Berbasis Binary

Dosen Pembimbing : **Muhammad Rudyanto Arief, S.T., M.T.**

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya.
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Dosen Pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta.
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi.

Yogyakarta, 30 Juni 2025

Yang Menyatakan,

Ahmad Hassan Rasyidi

KATA PENGANTAR

Dengan menyebut nama Allah SWT yang Maha Pengasih lagi Maha Penyayang, kami panjatkan puja dan puji syukur atas kehadirat-Nya, yang telah melimpahkan rahmat, serta hidayah, dan inayah-Nya sehingga saya dapat menyelesaikan penyusunan skripsi ini dengan judul "Implementasi Fuzzing menggunakan AFL++ untuk menemukan kerentanan pada perangkat lunak berbasis binary", Sholawat serta salam semoga tetap tercurahkan kepada junjungan Nabi Muhammad SAW yang senantiasa kita harapkan syafaatnya di hari akhir kelak.

Skripsi ini penulis ajukan untuk memenuhi syarat kelulusan pada mata kuliah Skripsi di Fakultas Ilmu Komputer, Program Studi Teknik Komputer, Universitas Amikom Yogyakarta. Adapun penyusunan skripsi ini juga merupakan salah satu syarat untuk memperoleh gelar Strata 1. Dalam penyusunan skripsi ini berbagai pihak telah memberikan dorongan, bantuan, serta masukan sehingga dalam kesempatan ini penulis menyampaikan terima kasih yang sebesar- besarnya kepada :

1. Ayah serta Ibu penulis yang senantiasa memberikan dukungan, do'a, dan motivasi untuk penulis sehingga dapat menyelesaikan pendidikan di Universitas Amikom Yogyakarta
2. Bapak Prof. Dr. M. Suyanto, M.M, selaku Rektor Universitas Amikom
3. Ibu Prof. Dr. Kusrini, M. Kom. selaku Dekan Fakultas Ilmu Komputer Universitas Amikom Yogyakarta.
4. Bapak Dr. Dony Ariyus, M.Kom selaku Kepala Prodi Teknik Komputer.
5. Bapak M. Rudyanto Arief, S.T, M.T selaku dosen pembimbing yang dengan sabar dan ikhlas telah meluangkan waktu dan memberikan ilmunya kepada penulis dalam memberikan petunjuk dan bimbingan sehingga penulis dapat menyelesaikan skripsi ini.
6. Kepada seseorang yang tidak kalah penting kehadirannya, Fravangastha Hangyang Levi Garniswaninghyun. Terima kasih telah menjadi bagian

dari proses perjalanan penulis untuk menyelesaikan skripsi. Berkontribusi baik tenaga, waktu, mendukung serta menghibur penulis dalam kesedihan, mendengarkan keluh kesah serta meyakinkan penulis untuk pantang menyerah sampai skripsi ini terselesaikan.

7. Rangga Wahyu Setiawan dan Adam Firdaus selaku teman dekat penulis yang selalu memberikan semangat agar skripsi saya dapat segera diselesaikan.
8. Semua pihak yang telah memberikan kontribusi, baik secara langsung maupun tidak langsung yang tidak dapat saya sebutkan satu per satu dalam penyelesaian skripsi ini. Terima kasih atas kesempatan dan dukungan yang telah diberikan selama proses penulisan skripsi ini. Semoga skripsi ini dapat memberikan manfaat dan kontribusi yang berarti bagi perkembangan ilmu pengetahuan di masa depan yang akan datang.

Yogyakarta, 24 Juni 2025
Penulis

Ahmad Hassan Rasyidi

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMBANG DAN SINGKATAN

AFL++	American Fuzzy Looping
CIA	Confidentiality, Integrity, Availability
GCC	GNU Compiler Collection
NIST	National Institute Standards and Technology
PCI-DSS	Payment Card Industry Data Security Standard
GDPR	General Data Protection Regulation
Asan	AddressSanitizer

DAFTAR ISTILAH

Fuzzing	: Inputan secara Acak
AddressSanitizer	: Deteksi error memori saat fuzzing
Buffer Overflow	: Input yang melewati batas Buffer Heap atau Stack
Integer Overflow	: Angka yang melampaui batas
Dynamic Testing	: Pengujian Security saat perangkat lunak berjalan
Static Testing	: Analisis kode Manual maupun Otomatis
Arbitrary Code Execution	: Eksekusi Kode Berbahaya

INTISARI

Keamanan perangkat lunak merupakan aspek yang sangat krusial dalam pengembangan perangkat lunak modern, terutama dengan semakin meningkatnya ketergantungan terhadap teknologi digital dalam berbagai sektor kehidupan. Kerentanan dalam perangkat lunak dapat dimanfaatkan oleh pihak tidak bertanggung jawab untuk melancarkan serangan yang mengancam prinsip-prinsip dasar keamanan informasi, yaitu kerahasiaan (Confidentiality), integritas (Integrity), dan ketersediaan (Availability). Penelitian ini bertujuan untuk mengetahui jumlah crash yang dihasilkan selama proses fuzzing, mengidentifikasi kerentanan keamanan dari crash yang ditemukan, serta mengevaluasi apakah seluruh crash tersebut benar-benar mengindikasikan adanya kerentanan atau hanya menyebabkan gangguan tanpa risiko eksploitasi. Fuzzing dilakukan pada sistem operasi Linux dengan perangkat lunak target yang telah dikompilasi menggunakan Instrumentasi dan Address Sanitizer (ASan). Hasilnya menunjukkan bahwa ditemukan sebanyak 76 file crash, di mana 6 di antaranya bersifat unik. Berdasarkan analisis, diketahui bahwa kerentanan yang ditemukan termasuk dalam kategori Stack Buffer Overflow. Kondisi ini membuka kemungkinan terjadinya Arbitrary Code Execution, terutama jika aplikasi dijalankan dengan hak akses tinggi. Penelitian ini menunjukkan bahwa sanitasi input yang ketat perlu diterapkan untuk mencegah eksploitasi. Diperlukan juga evaluasi menyeluruh terhadap fungsi-fungsi yang menangani input agar aplikasi lebih tahan terhadap serangan.

Kata kunci: Binary Fuzzing, American Fuzzy Looping, Buffer Overflow, AddressSanitizer, Keamanan Aplikasi.

ABSTRACT

Software security is a crucial aspect of modern software development, especially with the increasing reliance on digital technology in various sectors of life. Vulnerabilities in software can be exploited by irresponsible parties to launch attacks that threaten the basic principles of information security, namely confidentiality, integrity, and availability. This research aims to determine the number of crashes generated during the fuzzing process, identify security vulnerabilities from the crashes found, and evaluate whether all crashes actually indicate a vulnerability or only cause disruption without the risk of exploitation. Fuzzing was performed on a Linux operating system with target software that had been compiled using Instrumentation and Address Sanitizer (ASan). The results showed that a total of 76 crash files were found, of which 6 were unique. Based on the analysis, it is known that the vulnerabilities found fall into the Stack Buffer Overflow category. This condition opens the possibility of Arbitrary Code Execution, especially if the application is run with high access rights.This research shows that strict input sanitisation needs to be implemented to prevent exploitation. A thorough evaluation of the functions that handle inputs is also needed to make the application more resilient to attacks.

Keyword: *Binary Fuzzing, American Fuzzy Looping, Buffer Overflow, AddressSanitizer, Software Security*.